CHAPTER 19

# Inserting Graphics and Other Objects

**In This Chapter**

An introduction to graphics and multimedia file formats

The IMG element

The OBJECT element

Tips on using graphics effectively

Suggestions on providing alternatives for text-only browsers

What is a Web page without graphics? This chapter takes you through the basics of inserting graphics and multimedia effectively into your pages. The `IMG` and `OBJECT` elements enable you to insert images and other types of multimedia into your pages.

## A Quick Introduction to Graphics File Formats

Image files come in many formats. Fortunately, the Web only supports three formats: JPEG (Joint Photographic Experts Group), GIF (Graphics Interface Format, pronounced *jiff* by the inventor of the format, and *gif*, as in gift, by everyone else), and PNG (Portable Network Graphic). Initially, the Web only supported GIF images for inline viewing and JPEG images for out-of-line viewing. Today, you can use all three formats in the middle of your page. The Web uses these three types of graphics files because each format supports some type of compression.

### JPEG

JPEG compresses your image by removing redundant data. It maintains the complete color spectrum of your image. If you have a photo of yourself in front of a tree or a beautiful sunset, JPEG is a good compression choice. In the tree photo, the slightly inaccurate re-creation of the leaf pattern is of little consequence. Because JPEG preserves all the colors, your tree and sunset are as lovely as the original when compressed with JPEG.

### GIF

GIF compresses your image by removing colors down to the 256-color palate. As a result, your GIF photo of a sunset probably has stripes (where one color ends and the next closest color begins), and your tree is also less vibrant. If you have an architectural drawing in which the straightness of the lines matters more than the background shade of sepia, then GIF is the proper format. Because this format handles line art skillfully, use GIFs to save text images.

### PNG

PNG is the newest graphics file format supported on the Web. By using an exclusively designed algorithm, PNG provides the best overall quality. Only the newest browsers support PNG, however. If visitors view your page without the latest browser, they receive the broken-image graphic — unless you provide an alternate image.

### Creating a graphic image

Many available software packages create graphics. Chapter 37 discusses your options for creating images and goes into greater detail about selecting the appropriate format.

## Adding Inline Images with the IMG Element

Use the `IMG` element to add an image to your page simply. The `IMG` element includes everything you need in the start tag. `IMG` does not have content or end tags.

**Image** `<IMG>`

**Start Tag:** Required

**Content:** Empty

**End Tag:** Forbidden

**Attributes:** `id`, `class`: used in conjunction with style sheets

`lang`, `dir`: indicate language and text direction

`src`: URL of image

`alt`: short description; used for text-only browsers

`longdesc`: long description; used for text-only browsers

`title`: title of this element

`style`: define style here to override style sheet

`usemap`: URL of client-side image map

`ismap`: use server-side image map

`height`: height of image; use if overriding the actual image height

`width`: width of image; use if overriding the actual image width `align`, `hspace`, `vspace`, `border`: deprecated; should be specified in style sheet

`events`: see Chapter 48

The two most important (and required) attributes are `src` and `alt`. The `src` attribute specifies from where the image arrives. Even with an inline image, the image exists as a separate file, usually stored in your images directory. To support the shrinking number of text-only browsers, as well as the growing number of text-synthesizing browsers, include a useful `alt` attribute value. The `alt` attribute should contain a short, accurate description of the image's appearance. If you need a longer description, you should use the `longdesc` attribute.

`Height` and `width` serve two purposes. If you use an HTML editor to specify an image, the HTML editor automatically inserts the `height` and `width` attributes to facilitate quick page loading. In addition, if you want to load a small image and stretch it into a larger image, then use the `height` and `width` attributes.

Don't try to stretch a GIF image. Use JPEGs and PNGs for stretching instead.

`Usemap` and `ismap` are associated with image maps. Image maps are images with clickable areas — actually hyperlinks — under the image. Image maps are discussed fully in Chapter 39.

The following examples show how to use the `IMG` element:

```
<IMG class="logo" src="images/logo.gif" alt="Really Big Company
Logo">
```

This code inserts an image called `logo.gif` into your page. If you hold your mouse over the image using Internet Explorer 3 and above, or Netscape Navigator 4 and above, the words `Really Big Company Logo` pop up. The formatting — padding, borders, alignment, and text wrapping — of the image depends on your style sheet and the definition of the `IMG class` logo.

```
<IMG class="wrapped" src="images/stretch-box.gif"
alt="Concentric boxes" height="200" width="200">
```

The preceding code inserts an image called `stretch-box.gif`. The original image is 16×16 pixels, while the stretched image will be 200×200 pixels. If successful, stretching an image is a great way to minimize the load time of your page. The formatting of the image on the page depends on how the IMG class `"wrapped"` is defined in your style sheet.

# Multimedia

A growing number of inline and external multimedia formats work on the Web. Some multimedia files play in your browser without any special plug-ins, while others require plug-ins. Multimedia files can include video, sound, animation, VRML (Virtual Reality Modeling Language), QuickTime VR, and a number of other formats.

As the page designer, remember that visitors to your site may need special software to load your page properly. You should include links to any plug-ins required for your page. Some of the more common multimedia file formats follow:

| *Extension* | *Description* |
|---|---|
| .mov | movie |
| .wav | sound |
| .qtvr | QuickTime VR |
| .wrl | Worlds 3D VRML |
| .midi | sound |
| .dcr | Shockwave |
| .pdf | portable document format (Adobe) |

# Adding Inline Multimedia with the OBJECT Element

With more formats and each requiring different information, adding multimedia is trickier than adding an image. Add multimedia to your page using the OBJECT element. Currently, most images are inserted into your page using the IMG element, but even images will eventually be inserted using the OBJECT element.

**Object** <OBJECT>

**Start Tag:** Required

**Content:** Alternate text

Other OBJECT elements

One or more PARAM elements, description of what should appear

**End Tag:** Required

**Attributes:** id, class, lang, dir, events

declare: Boolean flag indicating the object is declared but should not be instantiated; if you use this attribute, you'll have to instantiate the object later with another OBJECT element

classid: URL of object's implementation; may be used either in conjunction with or instead of the data attribute, depending on the object

codebase: URL of base point for resolving relative references related to this object (specifically from the classid, data, and archive attributes)

codetype: content type of object specified in classid attribute in MIME format; saves time for a visitor downloading a content type that his or her browser can't render

data: URL of the object's data
type: MIME type of data specified in the data attribute; saves time for a visitor downloading a content type that his or her browser can't render

archive: list of URLs in space-delimited format that gives resources relative to the object's classid and data attributes; it may include the object's classid and data attributes; and the browser should preload the archive, which results in a quicker object load

standby: a message that the browser may show while loading the object

title: title of this element

style: define style here to override style sheet

usemap: URL of client-side image map

name: name for form submission

height: height of object; use if overriding the actual image height

width: width of object; use if overriding the actual image width

align, hspace, vspace, border: deprecated; should be speci hfgfied in style sheet

tabindex: tabbing order (for forms)

The OBJECT element can be very confusing. Fortunately, you will probably never need to use all attributes at the same time.

The OBJECT element can be nested into a sort of pecking order. Your browser uses the first instance of the nested OBJECT element that it supports. If you, as a Web developer, put the spiffiest items first, then the visitor sees the coolest multimedia that her browser can support without realizing she is missing any items, as shown in the upcoming example.

You want to include a model of a protein molecule for your cool chemistry site. The coolest model is the VRML version of the molecule, which is 3D and rotates and zooms at the visitor's discretion. Unfortunately, the visitor needs to have a computer that can handle 3D rendering and a VRML player installed. The next coolest way to show the model uses a QuickTime VR model of the molecule. This method enables the visitor to rotate the model without the ability to zoom and travel through the molecule. If the visitor can't even handle the QuickTime VR model, use an MPEG movie of the molecule bonding with another molecule. The last option is a GIF image of the molecule. For text-only browsers, settle for a short, text description of the molecule.

The following creates the nested OBJECT elements for this example:

```
<OBJECT classid="http://nmc.itc.virginia.edu/nlii/Grisham/
vrml/modelA1.wrl" codetype="application/wrl" height="250"
width="600">
  <OBJECT classid= "http://nmc.itc.virginia.edu/qtvr/atp_
synthase.qtvr" codetype="application/quicktime">
       <OBJECT data="synthase.mpeg" type="application/mpeg">
            <OBJECT data="atp_synthase.gif" type="image/gif">
                 You are missing the ATP Synthase molecule.
            </OBJECT>
       </OBJECT>
  </OBJECT>
</OBJECT>
```

# Tips on Using Images Effectively

Good Web designers often debate when and how to use images. The two major issues with regards to images on your page are download time and visual clutter. No hard and fast rules about using images exist, but the following guidelines help:

1. **Use the lowest color depth appropriate for your application.** Chapter 37 discusses this issue in greater depth. In simple terms, don't save your black-and-white image in 16.7 million colors. If your image uses only primary colors, save it with the least color-palette baggage to reduce download time tremendously.
2. **Use the smallest image appropriate.** What are you trying to communicate with the image? You need to have a certain level of detail in your image to make your point, but try to keep images small. Crop them tightly and leave the white space on the page using padding in the style sheet.

3. **Use text instead of a text image, if at all possible.** If the your font is widely available (meaning it comes installed with Windows 95/98/NT with a reasonable facsimile available on the Mac), then use text; if not, use an image of text. (See Figure 19-1.)

**Figure 19-1:** Using a text image is sometimes unavoidable.

4. **Keep images in the context of your page.** Make sure all your images contribute to the message of your page. Don't add images just because you like them. Avoid visual clutter by prioritizing the message of your page.

# Providing Alternatives for Text-Only Browsers

Only a small fraction of Web surfers have text-only browsers. Should you bother to mark up your page to accommodate them? Probably not, but you should include alternatives for other reasons. The Americans with Disabilities Act (ADA) requires all government sites to be accessible to the visually impaired.

What does the term *speech-synthesizing browser* mean to you? If you only think of the visually impaired, broaden your horizon. The biggest growth in speech-synthesizing browsers will come in alternative browser locations, such as your car! Because of this growing segment of the browser market, you should include alternatives to images and multimedia for text-only browsers.

How can you make your Web page friendly to speech synthesizing?

- **Use the alt attribute on every image.** Since speech-synthesizing browsers can't see your image, state your image contents.
- **Use text as the content of your OBJECT element.** Speech-synthesizing browsers skip the OBJECT element but process the content — the equivalent of alt text.
- **Never use the deprecated FONT element.** This element gives speech-synthesizing browsers fits. Use style sheets instead.
- **Use images only when absolutely necessary.** Use text instead of an image of text if possible. The image of text can be anti-aliased, but your ears-only visitors hear silence instead of your lovely image of text.

# From Here

Go to Chapter 37 and learn about creating graphics for the Web.

Jump to Chapters 40 through 43 and learn about creating multimedia for the Web, including sound, video, ActiveX controls, and Java.

Proceed to Chapter 20 to begin adding tables to your Web page..

# Summary

This chapter taught you how to add images and multimedia to your pages. You learned the `IMG` element and the dreaded `OBJECT` element. The examples provided with the `OBJECT` element should clarify its use. Finally, you learned about using images on your page effectively and accommodating text-only browsers.

✦ ✦ ✦